REMARKS

Claims 1-7, 9-10, and 17-19, and 21-34 are all the claims presently pending in the Application. Claims 8, 11-16, and 20 are canceled. Claims 9 and 17-25 are rejected as indefinite and claims 1-7, 9-10, and 17-19, and 21-25 stand rejected for informalities stand rejected on prior art grounds. New claims 26-34 have been added to claim additional features of the invention.

It is noted that any claim amendments are made to merely clarify the language of each claim, and not for distinguishing the invention over the prior art, narrowing the claims or for any statutory requirements of patentability. It is further noted that, notwithstanding any claim amendments made herein, Applicant's intent is to encompass equivalents of all claim elements, even if amended herein or later during prosecution.

Regarding the prior art rejections, claims 1-7, 9-10, and 17-19, and 21-25 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Robertazzi et al (Robertazzi), U.S. Patent No. 6,370,560.

The rejection is respectfully traversed in view of the following discussion.

I. THE 35 U.S.C. § 112, SECOND PARAGRAPH, REJECTION

Claims 9 and 17-25 have been rejected under 35 U.S.C. §112, second paragraph. Claim 9 has been amended to read "determining a revised estimate" in the

third line, which is referenced in the last line of the claim. Claims 17-19 and 21-25 have been amended, removing references to "kth functions" and "jth instances." The claims simply recite a "function" of an "instance" with antecedent baseis where appropriate. Claim 20 was canceled and therefore the rejection for this claim is moot.

In view of the foregoing, Applicant submit that claims 9, 17-19, and 21-25 are amended to overcome the Examiner's rejections. The Examiner is respectfully requested to reconsider and withdraw this rejection.

II. THE PRIOR ART REJECTION

The Examiner alleges that claims 1-25 are rejected under 35 U.S.C. § 103(a) as unpatentable over Robertazzi. Claims 8, 11-16, and 20 are canceled and therefore the rejection to these claims is moot. Applicant submits that there are elements of the claimed invention which are neither taught nor suggested by the Robertazzi.

THE ROBERTAZZI REFERENCE

Robertazzi alleges a "load sharing system which minimizes overall costs by assigning segments of a divisible load to distributed processor platforms based on the resource utilization cost of each processor platform." (Robertazzi, Abstract) The disclosure applies to "divisible loads" that can be "parallel processed." (Col. 3, line 55). The system performs load sharing operations based on the resource utilization cost of

each parallel processor over a network. (Col. 5, lines 17-22) The system reallocates a portion of the most expensive processor's load to a cheaper processor's load that can complete the execution by the finish time plus an incremental finish time. (Col. 9, lines 44-51) The system calculates the overall cost as a summation of the monetary cost /unit load of each processor platform and their associated data links multiplied by the fraction of the load or task processed on each respective processor. (Col. 11, lines 15-25) The goal is to optimize the resource utilization cost of computation of the distributed processes within a network using parallel processing. (Col. 1, lines 15-17).

On page 3 of the Office Action, the Examiner alleged that column 3, lines 1-8 and column 6, lines 18-36 disclose a method of "determining an amount of the processor resource available to be assigned." However, these sections of Robertazzi fail to teach or suggest a method to divide and change the computational load on an algorithm itself. Robertazzi's method merely divides parts of a fixed computational load among different processors on a cost-basis. It fails to teach or suggest "determining an amount of processing resources available to be assigned within a processor to functions of one or more algorithms," as recited in claim 1.

Robertazzi discloses a multiple processors system (e.g., parallel processing) and fails to teach or suggest determining processing resources "within a processor," as recited in claim 1. In Robertazzi, the "controller determines the resource utilization cost of a distributed processor. . . ." (col. 3, lines 4-6). "When a controller receives a divisible job from the job queue," the controller searches for ""the resource utilization

costs of any available distributed processor connected to the system's network which can process a portion of the load as well as the processor's data links cost." (col. 6, lines 20-24). First, Robertazzi's controller allocates divisible load jobs "among a plurality of distributed processor platforms." (col. 2, lines52-54). Robertazzi's goal is to use more than one processor for processing, whereas the present invention uses a novel and efficient method for processing within a single processor. The processors from Robertazzi are remote from one another, and Robertazzi even calls for using the Internet to link remote processors that "allows for the fast and cheap communication between processors irrespective of physical locations." (col. 2, lines 27-30).

Further, Robertazzi's disclosure is specifically limited to "divisible loads and tasks which *can be* parallel processed on multiple processors" (col. 3, lines 55-56) (emphasis Applicant's). As one skilled in the art knows, these are fixed loads that use a fixed amount of CPU cycles. Robertazzi's controller cannot modify the computational requirements of the loads "within a processor." That controller simply divides a fixed load into segments and uses "brute" force through optimal distribution of a load.

The claimed invention does not necessarily apply to algorithms that have constant computational requirements. The claimed invention defines and allocates processing resources to functions of one or more algorithms, and as one skilled in the art understands these functions are from an algorithm that allows modification of the computational requirements. The claimed invention changes the required computations to fit them within existing constrains of a processor in order to maximize the utilization of

individual processing units rather than re-distributing a fixed load to other processors as in Robertazzi.

Robertazzi's disclosure completely ignores the specifics of the algorithms it is "redistributing" the computation for. The present invention has to be aware of the specifics of the algorithms so it can capitalize on them and optimize their load requirements based on their achieved performance. Robertazzi's "divisible load jobs" does not depend upon what the algorithm does and how "well" the algorithm performs, it simply performs its purpose and consumes a certain amount of resources. Thus, Robertazzi's disclosure of parallel processing for fixed loads is for a different problem and a different purpose than the present invention and does not teach or suggest determining an amount of processing resources available to be assigned within a processor to functions of one or more algorithms, as recited in claim 1.

The Examiner further alleges column 6, lines 18-26 of Robertazzi discloses "determining an amount of the resource needed for each function waiting in the queue to execute" and that 5, lines 51-60 of Robertazzi discloses "allocating the available resources to the functions on a hierarchical priority scheme." However, these passages from Robertazzi merely describe distributing a "divisible load or task" from a job queue for processing by multiple, remote processors using "any conventional queuing scheme." Again, the goal of Robertazzi is "containing the current costs of available processors and their data links." (col. 6, lines 27-29) In Robertazzi, all of the load segments still have to be executed, but the controller determines how to redistribute the

load segments in the optimal cost-effective way. This is in complete contrast to the claimed invention where certain functions may not be executed within the same processor in order to optimize the processing resources within the processor.

The metrics of performance in Robertazzi is the dollar cost of computations, cost of links between computers, and possibly delay in processing. Stated plainly, Robertazzi does not manipulate resources within a processor itself This does not teach or suggest "determining an estimate of an amount of the processing resources needed for each function waiting in a queue to execute; and allocating the available processing resources within said processor to the functions based on an allocation scheme," as recited in claim 1. In the present invention, the metrics of performance is to what extent the end users could perceive that the method allocates processing resources for, and executes certain functions of, an algorithm at a time period before executing the remaining functions in a queue. The effect is that at any one time period, certain instance functions waiting in a queue may be executed and others may not. (see Application, p. 7, lines 6-20). Thus, if processing resources in a processor are inadequate, the present invention will not process all the functions of an algorithm, which would not be the case if enough resources were available. As one skilled in the art knows, this is only possible for algorithms that do not have fixed load requirements and can still perform satisfactorily when the algorithms do not perform all their operations that would have otherwise been performed under unlimited processing resources.

Determining the amount or processing resources in a processor for each function and allocating the resources to each function, as recited in claim 1, is not taught or suggested by Robertazzi because in that disclosure, once one of the processor reaches its processing limit there are no processing units remaining to process more load segments in order obtain its goal of optimizing costs. In contrast to Robertazzi, the claimed invention changes an algorithm's requirements so that the algorithm itself is influenced and changed. This method controls the load requirement in the processor rather than treating the load requirement as fixed and searching for additional processors in which to redistribute the load. Robertazzi is forced to increase the number of central processing units (CPUs) on different platforms in order to optimize the cost. That disclosure attempts to minimize the costs of processing but the result is an increase in the cost of using additional processors because the computational requirements of the load itself cannot be modified.

In other words, the segmented load in Robertazzi requires a fixed amount of CPU cycles and the only way to optimize those loads are to add additional CPUs, resulting in an increase in the size and cost of processing through redistribution. This is a complete contrast to the present invention that allocates the available processing resources within a processor to the functions, based on an allocation scheme. The present invention changes the load on a single processor without risking system performance and without requiring additional hardware. Robertazzi's disclosure would not result in an increase in channel density of processor in the present invention since

many more processing units would be required to process a full load. The claimed invention determines the amount of processing resources in a process, estimates the load requirements for each function of an algorithm, and then allocates resources to that specific function without a perceptible performance degradation by the end user. Thus, the cost and hardware required per channel is significantly reduced for performing a greater amount of processing. Once the processor's capacity is full, Robertazzi's method would shut down any further functions attempted to be processed in that processor, which is exactly the problem solved by the novel methods of the present invention.

Thus, the claimed invention is a significant advancement over any type of distributed processing on multiple paltforms. Robertazzi's method does not address and cannot achieve the performance on a single processor as the present invention. The present invention accomplishes controlled computational changes on algorithm functions that have a minimal effect on perceived performance.

Further, the Examiner admits that "Robertazzi does not specifically teach the use of estimating the amount of the resources needed," and that "it is well known that one of ordinary skill in the art at the time of the invention was made to include the option of estimating the amount to the existing system of Robertazzi." This is because "it would increase the accuracy of the amount of the resources needed thus improving overall planning of the system and resources." Applicant respectfully submits that the this is incorrect. One skilled in the art, reading Robertazzi, would estimate processing

requirements on fixed load segments for multiple, remote parallel processors. This is completely contrasted to the claimed invention, which determines "an estimate of an amount of the processing resources needed for each function waiting in a queue to execute," as recited in claim 1. As described, the claimed invention changes an algorithm's requirements so that the algorithm itself is influenced and changed.

The claimed method controls the load requirement in the processor rather than treating the load requirement as fixed and searching for additional processors in which to redistribute the load. As estimate of Robertazzi's method would force one skilled in the art to increase the number of central processing units (CPUs) on different platforms in order to optimize the cost. That disclosure attempts to minimize the costs of processing but the net result is actually an increase in the cost of using additional processors because the computational requirements of the load itself cannot be modified.

Thus, turning to the exemplary language of claim 1, there is no teaching or suggestion in the Robertazzi of a method of allocating processing resources to functions in a queue waiting to be executed, comprising:

determining an amount of processing resources available to be assigned within a processor to functions of one or more algorithms;

determining an estimate of an amount of the processing resources needed for each function waiting in a queue to execute; and

allocating the available processing resources within said processor to the

functions based on an allocation scheme." (emphasis Applicant's).

The Examiner further alleged Robertazzi discloses the inventions described in claims 2-7 and 9-10, which depend from claim 1. For the reasons stated above, the Examiner is respectfully incorrect. Each of these dependent claims recite aspects of the invention regarding the functions of the one or more algorithms described in claim 1. As described above, these algorithmic functions that are queued in a processor are fundamentally different than the fixed load segments described by Robertazzi.

The Examiner further alleged that column 5, lines 19-25 of Robertazzi discloses the elements of claim 17 and that one skilled in the art would know the amount of resources can be an estimate. However, these sections from Robertazzi are contrasted with the present invention in claim 17 for the reasons described above regarding claim 1. In claim 17, the present invention receives a "plurality of communication links into a processor, wherein each said communication link creates a separate instance for said processor to execute." The method estimates the amount of processing resources needed in the processor for each instance to execute in the same time period. Then, if processing resources are inadequate, the resources are allocated to the instances in an allocation scheme. In contrast to Robertazzi, the claimed invention changes the computational requirements of the instances so that the software instances themselves are influenced and changed. This method controls the instance requirement in the processor rather than the Robertazzi method of treating a segmented load requirement as fixed and searching for additional processors in which to redistribute the load.

For at least the reasons stated above, Applicant respectfully submits that the cited reference fails to teach or suggest every feature of claims 1 and 17. Claims 2-7, 9-10, 18-19, and 21-25 are patentable for the reasons above and for their dependence upon patentable claims. Therefore, the subject matters of claims 1-7, 9-10, 17-19, and 21-25 are fully patentable over the cited references.

Based on the foregoing, the Examiner is respectfully requested to reconsider and withdraw the rejection.

III. FORMAL MATTERS AND CONCLUSION

Applicant has revised the claims to overcome the Examiner's rejection for informalities. In view of the foregoing, Applicant submits that claims 1-7, 9-10, 17-19, and 21-34, all the claims presently pending in the Application, are patentably distinct over the prior art of record and are in condition for allowance. The Examiner is respectfully requested to pass the above Application to issue at the earliest possible time.

Should the Examiner find the Application to be other than in condition for allowance, the Examiner may contact the undersigned at the local telephone number listed below to discuss any other changes deemed necessary in a telephonic or personal interview.

The Commissioner is hereby authorized to charge any fees associated with this communication to Client's Deposit Account No. 20-0668.

Respectfully Submitted,

Date: 1-5-05

Kendal M. Sheets, Reg. No. 47,077
Joseph J. Zito, Reg. No. 32,076
Customer Number:23494
(301) 601-5010

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: the Commissioner for Patents, United Stated Patent and Trademark Office, PO Box 1450, Alexandria, VA 22313-1450 on January 5, 2005.

1-5-05

Kendal M. Sheets Date